

Q1. What will be the output of the program? [8 Marks]

```
1 //RunnableExample.java
2 class Count implements Runnable
3 {
4     Thread mythread ;
5     Count()
6     {
7         mythread = new Thread(this, "my runnable thread");
8         System.out.println("my thread created");
9         mythread.start();
10    }
11    public void run()
12    {
13        try
14        {
15            for (int i=0 ; i<5;i++)
16            {
17                System.out.println("Printing the count " + i);
18                Thread.sleep(1000);
19            }
20        }
21        catch (InterruptedException e)
22        {
23            System.out.println("my thread interrupted");
24        }
25        System.out.println("mythread run is over" );
26    }
27 }
28
29 class RunnableExample
30 {
31     public static void main(String args[])
32     {
33         Count cnt = new Count();
34         try
35         {
36             while(cnt.mythread.isAlive())
37             {
38                 System.out.println("Main thread will be alive till the child thread is live");
39                 Thread.sleep(1500);
40             }
41         }
42         catch (InterruptedException e)
43         {
44             System.out.println("Main thread interrupted");
45         }
46         System.out.println("Main thread run is over" );
47     }
48 }
```

C:\Program Files (x86)\Xinox Software\JCreatorV4\GE20...

```
my thread created
Main thread will be alive till the child thread is live
Printing the count 0
Printing the count 1
Main thread will be alive till the child thread is live
Printing the count 2
Main thread will be alive till the child thread is live
Printing the count 3
Printing the count 4
Main thread will be alive till the child thread is live
mythread run is over
Main thread is over
Press any key to continue..._
```

Q2. What will be the output of the program? [2 Marks]

```
1 import java.lang.*;  
2  
3 public class isAliveExample implements Runnable {  
4  
5     public void run() {  
6  
7         Thread t = Thread.currentThread();  
8  
9         System.out.println("status = " + t.isAlive());  
10    }  
11  
12    public static void main(String args[]) throws Exception {  
13  
14        Thread t = new Thread(new isAliveExample());  
15  
16        t.start();  
17  
18        t.join();  
19  
20        System.out.println("status = " + t.isAlive());  
21    }  
22 }
```



C:\Program Fi...

—



```
status = true  
status = false  
Press any key to continue...
```

